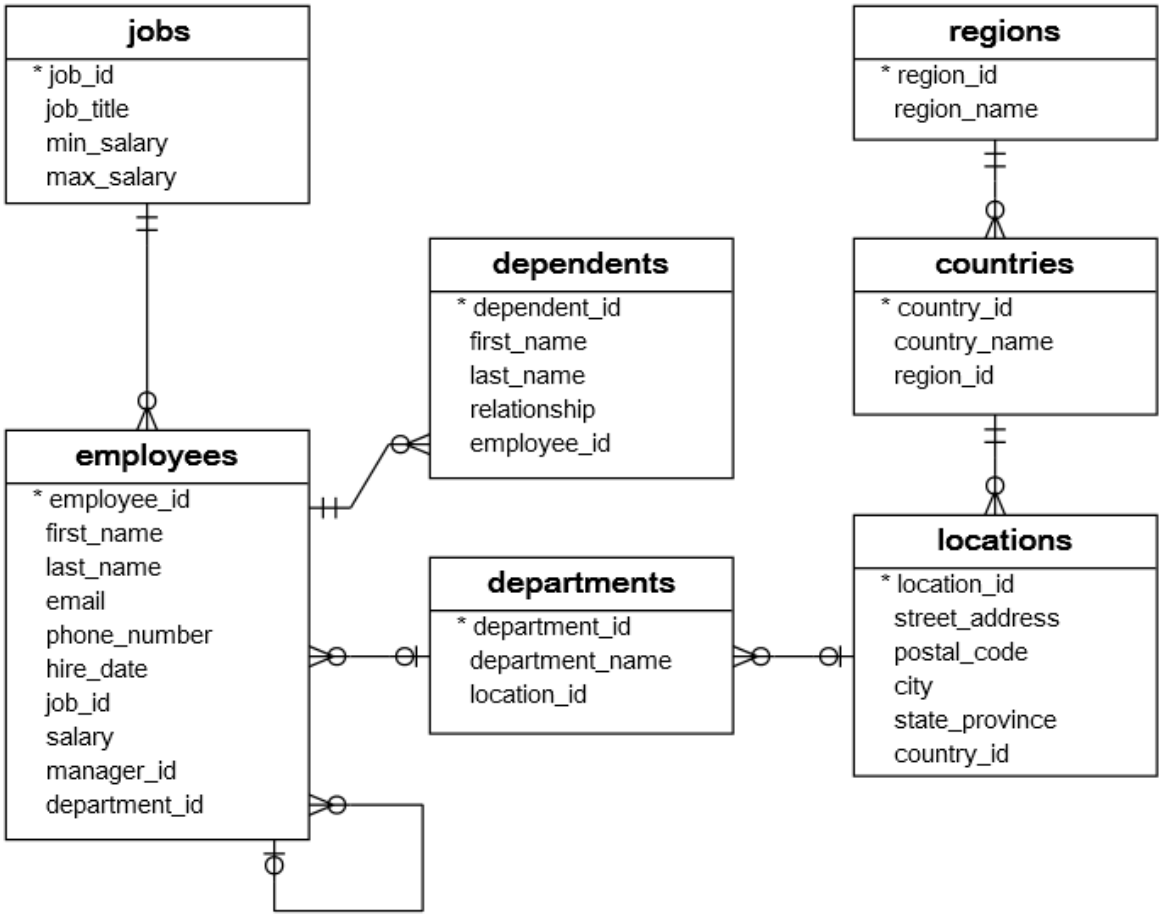
Aganitha AI mini Project

This jupyter notebook is hosted on an AWS EC2 instance here.
This notebook is running inside a docker container that is linked to a postgres container that contains a database named **hr_db** made from the script that was obtained from this webpage.

According to the source, the database is called **HR (hr_db in the docker container)** and it manages the HR data of small businessess. It has **7 different tables**. Now. let is dive a little deeper into the dataset and explore it.



Source of the image

To learn how to link a jupyter container to a postgres server and execute sql queries, refer to this and this.

In [1]:
```
import os
import psycopg2
import pgspecial
from sqlalchemy import create_engine
import pandas as pd
```

Loading the sql extension and connecting to the database in the postgres docker container.

hr_db is the name of the database and **postgres_server** is the name of the postgres docker container that is linked to the jupyter container that the notebook is running on.

In [2]:
```
%load_ext sql
```

In [3]:
```
%sql postgresql://postgres:password@postgres_server/hr_db
```

Out[3]:
```
'Connected: postgres@hr_db'
```

In [4]:
```
# creating the engine and connecting to the HR database.
engine = create_engine('postgresql://postgres:password@postgres_server/hr_db')
engine
```

Out[4]:
```
Engine(postgresql://postgres:***@postgres_server/hr_db)
```

Executing The SQL Queries in the database.

First, let us see all the tables in the database.

In [5]:
```
%%sql
\dt
```

Out[5]:
```
 * postgresql://postgres:***@postgres_server/hr_db
7 rows affected.
```

Schema	Name	Type	Owner
public	countries	table	postgres
public	departments	table	postgres
public	dependents	table	postgres
public	employees	table	postgres
public	jobs	table	postgres
public	locations	table	postgres
public	regions	table	postgres

In [32]:
```
# getting all the tablenames for later use
# sql query to list all tables in the database: https://stackoverflow.com/a/14730638
tnames = pd.read_sql_query('''SELECT table_name
  FROM information_schema.tables
 WHERE table_schema='public'
   AND table_type='BASE TABLE';''', engine)['table_name']
```

There are 7 tables in the database. Now, let us see the columns in each table.

First, let us see the columns in the jobs table.

In [43]:
```
# printing all the columns in each table
for tname in tnames:
    # the query
    query = '''SELECT
       table_name,
       column_name,
```

```
        data_type
    FROM
        information_schema.columns
    WHERE
        table_name = '{}';'''.format(tname)

    # executing the query
    res = pd.read_sql_query(query, engine)['column_name']

    # printing details
    print('For table {},'.format(tname))
    print('Columns:')
    print(*res.values, sep=', ')
    print('-'*100)
```

```
For table regions,
Columns:
region_id, region_name
----------------------------------------------------------------------------------------------------
For table countries,
Columns:
region_id, country_id, country_name
----------------------------------------------------------------------------------------------------
For table locations,
Columns:
location_id, street_address, postal_code, city, state_province, country_id
----------------------------------------------------------------------------------------------------
For table departments,
Columns:
department_id, location_id, department_name
----------------------------------------------------------------------------------------------------
For table jobs,
Columns:
job_id, min_salary, max_salary, job_title
----------------------------------------------------------------------------------------------------
For table employees,
Columns:
department_id, job_id, salary, manager_id, employee_id, hire_date, first_name, last_
name, email, phone_number
----------------------------------------------------------------------------------------------------
For table dependents,
Columns:
dependent_id, employee_id, first_name, last_name, relationship
----------------------------------------------------------------------------------------------------
```

Now, lets see the number of rows in each table.

In [57]:

```
for tname in tnames:
    # the query
    query = ''' SELECT
        COUNT(*)
    FROM
        {};
    '''.format(tname)

    # executing the query
```

```
    res = pd.read_sql_query(query, engine)

    # printing the details
    print('Number of rows in Table {}: {}'.format(tname, res['count'].values[0]))
```

```
Number of rows in Table regions: 4
Number of rows in Table countries: 25
Number of rows in Table locations: 7
Number of rows in Table departments: 11
Number of rows in Table jobs: 19
Number of rows in Table employees: 40
Number of rows in Table dependents: 30
```

Now, let us see the number of people working in each department.

In [84]:

```
query = ''' SELECT
        departments.department_name, COUNT(*) Employee_Count
    FROM
        employees
    JOIN
        departments
    ON
        employees.department_id = departments.department_id
    GROUP BY
        departments.department_name
    ORDER BY
        employee_count;
    '''

# executing the query
res = pd.read_sql_query(query, engine)
res
```

Out[84]:

	department_name	employee_count
0	Administration	1
1	Human Resources	1
2	Public Relations	1
3	Accounting	2
4	Marketing	2
5	Executive	3
6	IT	5
7	Finance	6
8	Purchasing	6
9	Sales	6
10	Shipping	7